Surrounding Word Sense Model for Japanese All-words Word Sense Disambiguation

Kanako KOMIYA [1] **Yuto SASAKI** [2] **Hajime MORITA** [3]
Minoru SASAKI [1] **Hiroyuki SHINNOU** [1] **Yoshiyuki KOTANI** [2]
[1]Ibaraki University [2]Kyoto University [3]Tokyo University of Agriculture and Technology
kanako.komiya.nlp@vc.ibaraki.ac.jp, 50010268030@st.tuat.ac.jp
morita@nlp.ist.i.kyoto-u.ac.jp, minoru.sasaki.01@vc.ibaraki.ac.jp
hiroyuki.shinnou.0828@vc.ibaraki.ac.jp, kotani@cc.tuat.ac.jp

Abstract

This paper proposes a surrounding word sense model (SWSM) that uses the distribution of word senses that appear near ambiguous words for unsupervised all-words word sense disambiguation in Japanese. Although it was inspired by the topic model, ambiguous Japanese words tend to have similar topics since coarse semantic polysemy is less likely to occur than that in Western languages as Japanese uses Chinese characters, which are ideograms. We thus propose a model that uses the distribution of word senses that appear near ambiguous words: SWSM. We embedded the concept dictionary of an Electronic Dictionary Research (EDR) electronic dictionary in the system and used the Japanese Corpus of EDR for the experiments, which demonstrated that SWSM outperformed a system with a random baseline and a system that used a topic model called Dirichlet Allocation with WORDNET (LDAWN), especially when there were high levels of entropy for the word sense distribution of ambiguous words.

1 Introduction

This paper proposes a surrounding word sense model (SWSM) for unsupervised Japanese all-words Word Sense Disambiguation (WSD). SWSM assumes that the sense distribution of surrounding words varies according to the sense of a polysemous word.

For instance, a word “可能性” (possibility) has three senses according to the Electronic Dictionary Research (EDR) electronic dictionary (Miyoshi et al., 1996):

(1) The ability to do something well

(2) Its feasibility

(3) The certainty of something happenings

Although sense (3) is the most frequent in the prior distributions, sense (1) will be more likely when the local context includes some concepts like “人間” (man) or “誰々の” (someone’s). It is challenging in practice to accurately learn the difference in the senses of surrounding words in an unsupervised manner, but we developed an approximate model that took conditions into consideration.

SWSM is a method for all-words WSD inspired by the topic model (Section 2). It treats the similarities of word senses using WORDNET-WALK and it generates word senses of ambiguous words and their surrounding words (Section 3). First, SWSM abstracted the concepts of the concept dictionary (Section 4) and calculated the transition probabilities for priors (Section 5). Then it estimated the word senses using Gibbs Sampling (Section 6) . Our experiments with an EDR Japanese corpus and a Concept Dictionary (Section 7) indicated that SWSM was effective for Japanese all-words WSD (Section 8) . We discuss the results (Section 9) and concludes this paper (Section 10) .

2 Related Work

There are many methods of all-words WSD. Pedersen et al. (2005) proposed calculation of the semantic relatedness of the word senses of ambiguous words and their surrounding words. Some papers have reported that methods using topic models (Blei et al., 2003) are most effective. Boyd-Graber et al. (2007) proposed a model, called Latent Dirichlet Allocation with WORDNET (LDAWN), which was a model where the probability distributions of words that the topics had were replaced with a word generation process on WordNet: WORDNET-WALK. They ap-

29th Pacific Asia Conference on Language, Information and Computation pages 35 - 43
Shanghai, China, October 30 - November 1, 2015

plied the topic model to unsupervised English all-words WSD. Although Guo and Diab (2011) also used the topic model and WordNet, they also used WordNet as a lexical resource for sense definitions and they did not use its conceptual structure. They reported that the performance of their system was comparable with that reported by Boyd-Graber et al.

There has been little work, on the other hand, on unsupervised Japanese all-words WSD. As far as we know, there has only been one paper (Baldwin et al., 2008) and there have been no reported methods that have used the topic model. We think this is because ambiguous words in Japanese tend to have similar topics since coarse semantic polysemy is less likely to occur compared to that with Western languages as Japanese uses Chinese characters, which are ideograms. In addition, Guo and Diab (2011) reported that *in word sense disambiguation (WSD), an even narrower context was taken into consideration*, as Mihalcea (2005) had reported. Therefore, we assumed that the word senses of the local context are differentiated depending on the word sense of the target word like that in supervised WSD. SWSM was inspired by LDAWN, it thus uses WORDNET-WALK and Gibbs sampling but it does not use the topics but the word senses of the surrounding words. We propose SWSM as an approach to unsupervised WSD and carried out Japanese all-words WSD.

3 Surrounding Word Sense Model

SWSM uses the distribution of word senses that appear near the target word in WSD to estimate the word senses assuming that the word senses of the local context are differentiated depending on the word sense of the target word. In other words, SWSM estimates the word sense according to $p(\boldsymbol{s}|\boldsymbol{w})$, which is a conditional probability of a string of senses, $\boldsymbol{s}$, given a string of words $\boldsymbol{w}$.

SWSM involves three assumptions. First, each word sense has a probability distribution of the senses of the surrounding words. Second, when c_i denotes the sense string of the surrounding words of the target word w_i, the conditional probability of c_i given w_i is the product of the those of the senses in c_i given w_i. For example, when w_i is "可能性" (possibility) and its surrounding words are "両者" (both sides) and "人間" (human), $c_i = (s_{both}, s_{human})$ and $P(c_i|s_{possibility}) = P(s_{both}|s_{possibility})P(s_{human}|s_{possibility})$ are defined where $s_{possibility}$, s_{both}, and s_{human} denote word senses of "可能性" (possibility), "両者" (both sides), and "人間" (human). Finally, each polyseme has a prior distribution of the senses. Given these assumptions, SWSM calculates the conditional probability of $\boldsymbol{s}$ that corresponds to $\boldsymbol{w}$, under the condition where $\boldsymbol{w}$ is observed as:

$$P(\boldsymbol{s}, \boldsymbol{c}|\boldsymbol{w}) = \prod_{i=1}^{N} P(s_i|w_i)P(c_i|s_i, \boldsymbol{w}), \quad (1)$$

where $\boldsymbol{c}$ denotes the string of c_i and N denotes the number of all the words in the text. The initial part on the right is the probability distribution of the word sense of each word and the last part is that of the senses of the surrounding words for each word sense. We set the Dirichlet distribution as their prior.

The final equation considering prior is described using the following parameters:

$$P(\boldsymbol{s}, \boldsymbol{c}, \boldsymbol{\theta}, \boldsymbol{\phi}|\boldsymbol{w}, \gamma_k, \tau_j) =$$

$$\prod_{k=1}^{W} P(\theta_k|\gamma_k) \prod_{j=1}^{S} P(\phi_j|\tau_j) \prod_{i=1}^{N} P(s_i|\theta_{w_i}) P(c_i|\phi_{s_j}, \boldsymbol{w}), \quad (2)$$

where W denotes the number of words, S denotes the number of senses, θ_k denotes the probability distribution of the senses of word k, and ϕ_j denotes the probability distribution of the word senses surrounding word sense j. θ_k and ϕ_j are the parameters of the multinomial distribution. γ and τ are the parameters of the Dirichlet distribution.

Eq. (2) is the basic form. We replace $\boldsymbol{\phi}$, the probability distribution of each sense, with the generation process by using the WORDNET-WALK of the concept dictionary. The WORDNET-WALK in this work does not generate words but word senses using a hyper-transition probability parameter, $S\alpha$. We set α according to the senses to differentiate the sense distribution of the surrounding words before training. By doing this, we can determine which sense in the model corresponds to the senses in the dictionary.

SWSM estimates the word senses using Gibbs sampling as:

(1) Pre-processing

1 Abstract the concepts in the concept dictionary

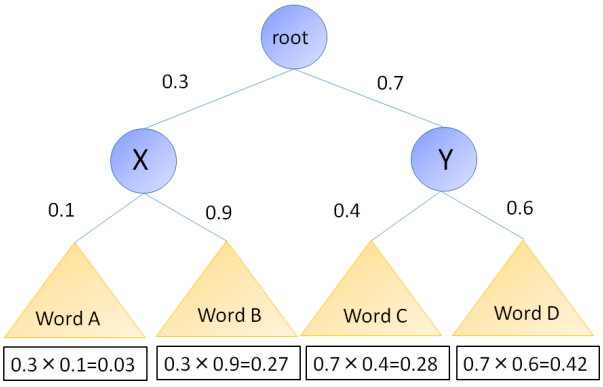

Figure 1: Example of WORNET-WALK

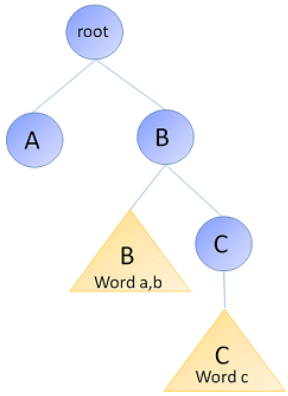

Figure 2: Addition of Word Sense Nodes

2 Calculate the transition parameters using the sense frequencies

(2) Training: Gibbs sampling to estimate the word senses

4 Concept Abstraction

SWSM obtains the sense probability of the surrounding words using WORDNET-WALK. WORDNET-WALK involves the generation process, which represents the probabilistic walks over the hierarchy of conceptual structures like WordNet. Figure 1 shows the easy example of the generation probabilities of words by WORDNET-WALK. When circle nodes represent concepts and triangle nodes represent words of leaf concepts ,i.e., X and Y, and numbers represent the transition probabilities, the generation probabilities of words A, B, C, and D are 0.03, 0.27, 0.28, and 0.42. LDAWN calculated the probabilities of word senses using the transition probability from the root node in a concept dictionary. WORDNET-WALK generated words in (Boyd-Graber et al., 2007) but our WORDNET-WALK generates word senses. However, the word senses sometimes do not correspond to leaf nodes but to internal nodes in our model and that causes a problem: the sum of the probabilities is not one. Thus, we added leaf nodes of the word senses directly below the internal nodes of the concept dictionary (c.f. Figure 2).

Concept abstraction involves the process by which hyponym concepts map onto hypernym concepts. Most concepts in a very deep hierarchy are fine grained like the "Tokyo University of Agriculture and Technology" and "Ibaraki University" and they should be combined together like "university" to avoid the zero frequency problem. Thus, SWSM combines semantically similar concepts in the concept dictionary.

Hirakawa and Kimura (2003) reported that they compared three methods for concept abstraction, i.e, flat depth, flat size, and flat probability methods, by using the EDR concept dictionary, and the flat probability method was the best. Therefore, we used the flat probability method for concept abstraction.

The flat probability method consists of two steps. First, there is a search for nodes from the root node in depth first order. Second, if the concept probability calculated based on the corpus is less than a threshold value, the concept and its hyponym concepts are mapped onto its hypernym concept.

We employed the methods of (Ribas, 1995) and (McCarthy, 1997) to calculate the concept probability. Ribas (1995) calculated the frequency of sense s as:

$$freq(s) = \sum_{w} \frac{|senses(w) \in U(s)|}{|senses(w)|} count(w), \quad (3)$$

where $senses(w)$ denotes the possible senses of a word w, $U(s)$ denotes concept s and its hyponym concepts, and $count(w)$ denotes the frequency of word w. This equation weights $count(w)$ by the ratio of concept s and its hyponym concepts in all the word senses of w. probability $P(s_i)$ was calculated as:

$$P(s_i) = \frac{freq(s_i)}{N}, \quad (4)$$

where N denotes the number of word tokens.

Figure 3 demonstrates the example of the conceptual structure[1]. The nodes A∼F represent the

[1]The leaf concepts below C, D, E, and F are omitted.

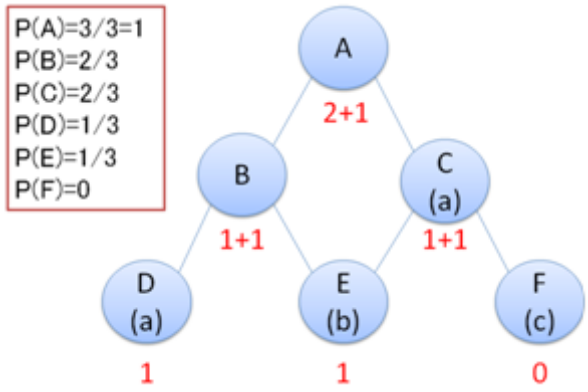

Figure 3: Example of Concept Structure

concepts and (a)∼(c) represent the words, which indicates that word (a) is a polyseme that have two word senses, i.e., C and D. When word (a) appeared twice and word (b) appeared once, the probabilities are as illustrated in Figure 3. Note that C and D share the frequencies of word (a).

A Turing estimator (Gale and Sampson, 1995) was used for smoothing with rounding of the weighted frequencies.

Concept abstraction sometimes causes a problem where some word senses of a polyseme are mapped onto the same concept. The most frequent sense in the corpus has been chosen for the answer in these cases.

5 Transition Probability

SWSM differentiates the sense distribution of the surrounding words of each target word before training using α : the transition probability parameter. As our method is an unsupervised approach, we cannot know the word senses in the corpus. Therefore, SWSM counts the frequencies of all the possible word senses of the surrounding words in the corpus. That is, if there are polysemes A and B in the corpus and B is a surrounding word of A, SWSM counts the frequencies of the senses by considering that all the senses of B appeared near all the senses of A. That makes no difference in the sense distributions of A; however, if there is another polyseme or a monosemic word, C, and a sense of C is identical with a sense of A, the sense distributions of A will be differentiated by counting the frequencies of the senses of C. As this example indicates, SWSM expects that words that have an identical sense, like A and C, have similar local contexts.

SWSM uses these counted frequencies to calculate the transition parameter α so that the transition probabilities to each concept are proportional to the word sense frequencies of the surrounding words. We calculate α_{s_i,s_j}, i.e., the transition probability from hypernym s_i to hyponym s_j, like that in (Jiang and Conrath, 1997) as:

$$\alpha_{s_i,s_j} = P(s_j|s_i) = \frac{P(s_i,s_j)}{P(s_i)} = \frac{P(s_j)}{P(s_i)}. \quad (5)$$

In addition, probability $P(s_i)$ is calculated as:

$$P(s_i) = \frac{freq(s_i)}{N}, \quad (6)$$

where $freq(s_i)$ denotes the frequency of sense s_i.

Moreover, $freq(s_i)$ is calculated like that in (Resnik, 1995):

$$freq(s_i) = \sum_{w \in words(s_i)} count(w). \quad (7)$$

Here, $words(s_i)$ denotes a concept set that includes s_i and its hyponyms, and N denotes the number of the word tokens in the corpus. However, the probability that Eq. (7) will have a problem, i.e., the sum of the transition probabilities from a concept to its hyponyms is not one. Thus, we calculate the probability by considering that the same concept that follow a different path is different:

$$freq(s_i) = \sum_{s_j \in L(s_i)} path(s_i,s_j) \sum_{w \in words(s_i)} count(w), \quad (8)$$

where $path(s_i, s_j)$ denotes the number of the paths from concept s_i to its hyponym s_j and $L(s_i)$ denotes the leaf concepts below s_i. Consequently, the transition probability can be calculated by dividing the frequencies of the hyponym by that of its hypernym.

When word (a) appeared twice and word (b) appeared once, the transition probability from A to B, i.e., $\alpha_{A,B}$ is 1/2 because the frequencies of A and B are six [2] and three in Figure 3.

Here, $p(path_{s_l})$, i.e., a transition probability of an arbitrary path from the root node to a leaf concept, $path_{s_l}$, is:

$$p(path_{s_l})$$

[2]It is sum of twice from path ABD (a), twice from path AC (a), once from path ABE (b), and once from path ACE (b).

$$= \frac{freq(c_1)}{freq(s_{root})}\frac{freq(c_2)}{freq(c_1)}\cdots\frac{freq(c_n)}{freq(c_{n-1})}\frac{freq(s_l)}{freq(c_n)}$$

$$= \frac{freq(s_l)}{freq(s_{root})}, \quad (9)$$

where $c_1c_2\ldots c_n$ denote the concepts in $path_{s_l}$. Therefore, when we set the frequency of the word sense frequencies of s_l, the surrounding words, as $freq(s_l)$, $p(path_{s_l})$ are proportional to the frequency.

We eventually used the following transition probability parameter to avoid the zero frequency problem:

$$S_a\alpha_a + S_b\alpha_b^s, \quad (10)$$

where α_a denotes a transition probability parameter where all the leaf nodes have the same amount probability and α_b^s denotes the transition probability parameter that is pre-trained using the above equations. S_a and S_b are constant numbers to control the effect of pre-processing.

The transition probability parameter where all the leaf nodes have the same amount probability, α_a, is calculated by assuming that the frequencies of all the leaf nodes are as follows. [3]

$$freq(s_l) = \frac{1}{path(s_{root}, s_l)} \quad (11)$$

6 Sense Estimation using Gibbs Sampling

SWSM estimates the word sense, s, using Gibbs sampling (Liu, 1994). As described in Section 3, the conditional probability of the model is in Eq. (12).

$$P(\boldsymbol{s}, \boldsymbol{c}, \boldsymbol{\theta}, \boldsymbol{\phi}|\boldsymbol{w}) =$$

$$\prod_{k=1}^{W} P(\theta_k|\gamma_k)\prod_{j=1}^{S} P(\phi_j|\tau_j)\prod_{i=1}^{N} P(s_i|\theta_{w_i})P(c_i|\phi_{s_j},\boldsymbol{w}) \quad (12)$$

We calculate the conditional distribution that is necessary for sampling. We regard variants except those for word w_i as constant numbers. The probability distribution, $\boldsymbol{\phi}$, of the word sense is actually replaced by WORDNET-WALK in the word sense generation process and it will have plural multinomial distributions of the transitions to the hyponym concepts.

We calculated the conditional distribution $P(s_i, c_i|\boldsymbol{s_{-i}}, \boldsymbol{c_{-i}}, \boldsymbol{w})$ as:

$$P(s_i = x, c_i = \boldsymbol{y}|\boldsymbol{s_{-i}}, \boldsymbol{c_{-i}}, \boldsymbol{w})$$

$$\propto (n_{w_i,x}^{-i}+\gamma)\cdot\prod_{j=1}^{|\boldsymbol{y}|}\frac{(n_{x,y_j}^{-i}+m_y(j,y_i)+\tau_{x,y_j})}{\sum_{sen}(n_{x,sen}^{-i}+\tau_{x,sen})+(j-1)}, \quad (13)$$

where x and $\boldsymbol{y}$ correspond to the real values of word sense s_i and the vector of the word senses of the surrounding words, c_i. $n_{w_i,x}^{-i}$ denotes the number of x, i.e., the word senses that are assigned to word w_i except for the i_{th} variate, which is the sampling target now. n_{x,y_i}^{-i} denotes the frequency where y_j appears around word sense x except for the i_{th} variate. $m_y(j, y_j)$ is the frequency where word sense y_j appear before the j_{th} surrounding word sense in $\boldsymbol{y}$ and it can be ignored if y_j appeared once in $\boldsymbol{y}$. We approximately and determinately assign the sequence of the word senses to $\boldsymbol{y}$, calculate each probability of s_i, and determine s_i, i.e., the word sense that corresponds to word w_i.

If the probability distributions of word senses are replaced with WORDNET-WALK, the last part of the right side of Eq. (13) will also be replaced. When $r_{j,0}, r_{j,1}, \ldots, r_{j,l}$ denotes the path from the root concept of word sense y_j in $\boldsymbol{y}$, we obtain Eq. (14) by calculating the following values of all combinations from the root concept for all word senses, and summing them.

$$\prod_{j=1}^{|y|}\prod_{p=1}^{l-1}\{T_{x,r_{j,p},r_{j,p+1}}^{-i} + m_y(j, r_{j,p}, r_{j,p+1}) + S_a\alpha_{a,r_{j,p},r_{j,p+1}} + S_b\alpha_{b,r_{j,p},r_{j,p+1}}^{x}\} / \{\sum_r (T_{x,r_{j,p},r}^{-i}+m_y(j,r_{j,p},r)+S_b\alpha_{b,r_{j,p},r}^{x})+S_a\}, \quad (14)$$

where $T_{x,r_{j,p},r_{j,p+1}}^{-i}$ denotes the frequency where the word sense of the surrounding words of word sense x pass the link from concept $r_{j,p}$ to concept $r_{j,p+1}$ except for the i_{th} variate. $m_y(j, r_{j,p}, r_{j,p+1})$ denotes the frequency where the link from concept $r_{j,p}$ to concept $r_{j,p+1}$ is passed before the j_{th} path. The value of T_{s_i} should be updated after word sense s_i is assigned. Thus, the paths of the word senses of the surrounding words are necessary. This time, we assign values proportional to each probability to each path. When $path_1$,$path_2$,

[3] The reason we did not set the frequencies of all the leaf nodes to one ($freq(s_l) = 1$) is as follows. If so, all the probabilities of all the paths from the root node to each leaf node would have been the same. However, the more paths from the root node a leaf node has, the higher the probability the leaf node will have. We used Eq.(11) so that all the leaf nodes would have the same probability.

$\cdots$,$path_n$ denote the paths from the root concept to word sense $c_{i,j}$, i.e., a word sense of surrounding words c_i of word sense s_i, we added following value to $T_{s_i,path_k}$, which is the frequency where a link in $path_k$ is passed, for each word sense $c_{i,j}$.

$$\frac{P(path_k|s_i)}{\sum_{l=1}^{n} P(path_l|s_i)} \tag{15}$$

The probability $p(path_k|s_i)$ is as follows, when $r_1, r_2, \cdots, r_l$ denote the concepts that $path_k$ follows.

$$P(path_k|s_i) = \sum_{p=1}^{l-1} \frac{T^{-i}_{s_i,r_p,r_{p+1}} + S_a \alpha_{a,r_p,r_{p+1}} + S_b \alpha^{s_i}_{b,r_p,r_{p+1}}}{\sum_r (T^{-i}_{s_i,r_p,r} + S_b \alpha^{s_i}_{b,r_p,r}) + S_a} \tag{16}$$

Concepts that have many paths from the root concept are concepts that have many properties. Thus, we can view these cases as that of an appearance of word sense $c_{i,j}$ that was assigned to multiple properties.

Algorithm 1 demonstrates the algorithm of one iteration in Gibbs Sampling of SWSM. Note that x and $\boldsymbol{y}$ are sampled according to Eq. (13) where the last part on the right side is replaced with Eq. (14) and each $T_{s_i,path_k}$ is updated with Eq. (15).

Algorithm 1 Processes of One Iteration in Gibbs Sampling of SWSM

Require: Disambiguate the word sense s_i in text
  for each word w_i in text **do**
    $n_{w_i,s_i} \Leftarrow n_{w_i,s_i} - 1$
    for each word sense $c_{i,j}$ in c_i **do**
      for each path $path_k$ for $c_{i,j}$ **do**
        $T_{s_i,path_k} \Leftarrow T_{s_i,path_k} - \frac{P(path_k|s_i)}{\sum_{l=1}^{n} P(path_l|s_i)}$
      end for
    end for
    $c_i \Leftarrow \boldsymbol{y}$
    $s_i \Leftarrow x$
    $n_{w_i,s_i} \Leftarrow n_{w_i,s_i} + 1$
    for each word sense $c_{i,j}$ in c_i **do**
      for each path $path_k$ for $c_{i,j}$ **do**
        $T_{s_i,path_k} \Leftarrow T_{s_i,path_k} + \frac{P(path_k|s_i)}{\sum_{l=1}^{n} P(path_l|s_i)}$
      end for
    end for
  end for

7 Data

We used the Japanese word dictionary, the concept dictionary, and the Japanese corpus of the second version of the EDR electronic dictionary. All the nouns and verbs that could be followed from the root node in the concept dictionary were used for the experiments. In addition, we added some nouns by deleting "する (suru, the suffix that means do)" from nominal verbs, to the concept dictionary. Consequently, the concept dictionary included 263,757 words and 406,710 leaf concepts, and 199,430 leaf concepts in them were used for the experiments. The internal nodes that were used for the experiments were 203,565 concepts. Most of the concepts that were not used were those that had no links to Japanese words. In addition, the concept dictionary included 13,846 concepts and 6,905 leaf concepts after concept abstraction. The threshold value we used was 5.0×10^{-5}.

The Japanese corpus consisted of seven sub-corpora: the Nikkei, the Asahi Shimbun, AERA, Heibonsha World Encyclopedia, Encyclopedic Dictionary of Computer Science, Magazines, and Collections. They were annotated with word sense tags that were the concepts in the concept dictionary. Table 1 summarizes the numbers of documents and word tokens according to the type of text. The documents in this corpus only consisted of one sentence.

Type of Text	Docs	Word tokens
The Nikkei	5,018	121,301
The Asahi Shimbun	91,400	2,272,555
AERA	49,589	1,183,897
Heibonsha World Encyclopedia	10,072	284,059
Encyclopedic Dictionary of Computer Science	13,578	357,607
Magazines	21,199	528,452
Collections	16,946	368,285

Table 1: Summary of Sub-corpora.

We used the Nikkei for evaluation. The other six sub-corpora were used for pre-processing in an unsupervised manner. The EDR Japanese corpus did not include the basic forms of words. Thus we used a morphological analyzer, Mecab[4], to identify the basic forms of words in the corpus.

Shirai (2002) set up the three difficulty classes listed in Table 2. Tables 7 and 3 indicate the number of word types, noun tokens, and verb tokens according to difficulty and the average polysemy

[4]https://github.com/jordwest/mecab-docs-en

of target words according to difficulty. Only words that appeared more than four times in the corpus were classified based on difficulty.

Difficulty	Entoropy
Easy	$E(w) < 0.5$
Normal	$0.5 \leq E(w) < 1$
Hard	$1 \leq E(w)$

Table 2: Difficulty of disambiguation

Difficulty	Word types	Tokens(N)	Tokens(V)
All	4,822	12,149	6,199
Easy	399	3,630	1,723
Normal	337	2,929	1,541
Hard	105	1,028	1,196

Table 3: Types and tokens of words according to difficulty

Difficulty	Noun polysemy	Verb polysemy
All	4.2	5.5
Easy	3.9	4.0
Normal	4.4	5.3
Hard	8.6	10.3

Table 4: Average polysemy of target words according to difficulty

8 Result

We used nouns and independent verbs in a local window whose size was $2N$ except for marks, as the surrounding words. We set $N = 10$ in this research. In addition, we deleted word senses that appeared only once through pre-processing.

We performed experiments using the nine settings of the transition probability parameters: $S_a = \{1.0, 5.0, 10.0\}$ and $S_b = \{10.0, 15.0, 20.0\}$ in Eq.(10). We set the hyper-parameter $\gamma = 0.1$ in Eq.(2) for all experiments. Gibbs sampling was iterated 2,000 times and the most frequent senses of 100 samples in the latter 1,800 times were chosen for the answers. We performed experiments three times per setting for the transition probability parameters and calculated the average accuracies.

Table 4 summaries the results. It includes the micro- and macro-averaged accuracies of SWSM for the nine settings of the parameters, those of the random baseline, and those of LDAWN [5]. The experiments for the random baseline were performed 1,000 times. The best results are indicated in boldface.

S_a	S_b	micro	macro
1	10	38.91%	**42.58%**
5	10	38.67%	42.42%
10	10	37.62%	42.37%
1	15	39.20%	42.43%
5	15	38.23%	42.29%
10	15	38.41%	42.17%
1	20	37.78%	42.26%
5	20	**39.60%**	42.09%
10	20	36.67%	42.04%
Random baseline		30.97%	36.63%
LDAWN		36.12%	42.51%

Table 5: Summary of result

The table indicates that our model, SWSM, was better than both the random baseline and LDAWN. Although the macro-averaged accuracies of LDAWN were better than those of SWSM except when $S_a = 1$ and $S_b = 10$, both the micro- and macro-averaged accuracies of SWSM outperformed those of LDAWN when $S_a = 1$ and $S_b = 10$.

Tables 5 and 6 summarize the micro-averaged accuracies of all words and the macro-averaged accuracies of all words. SWSM1 and SWSM2 in these tables denote the SWSMs with the setting when the best macro-averaged accuracy for all words was obtained ($S_a = 1$ and $S_b = 10$) and with the setting when the best micro-averaged accuracy for all words was obtained ($S_a = 5$ and $S_b = 20$). The best results in each table are indicated in boldface. These tables indicate that SWSM1 or SWSM2 was always better than both

[5] The best results for the 13 settings. We changed the number of topics and the scale parameters according to (Boyd-Graber et al., 2007). In addition, we tested that the effect of the size of a text, a sentence, or a whole daily publication because a document only consisted of a sentence in our Japanese corpus and there was no clues that indicated to what article the sentence belonged. Furthermore, we tested two kinds of transition probabilities, those that used priors and those where all the leaf nodes had the same amount probability. The best was the setting where there were 32 topics, scale parameter S was 10, the text size was a sentence, and the transition probabilities were those where all the leaf nodes had the same amount probability. The details are similar to those in (Sasaki et al., 2014). However, we performed the experiments three times and calculated the accuracies but they only performed the experiments twice.

the random baseline and LDAWN.

Method	All	Easy	Normal	Hard
Random	30.97	33.01	29.35	13.47
LDAWN	36.12	42.06	30.66	13.52
SWSM1	38.91	46.87	**33.44**	19.92
SWSM2	**39.60**	**48.90**	32.85	**23.95**

Table 6: Micro-averaged accuracies for all words (%)

Method	All	Easy	Normal	Hard
Random	36.63	36.91	32.09	16.03
LDAWN	42.51	44.65	34.83	17.80
SWSM1	**42.58**	**44.78**	**36.38**	**21.06**
SWSM2	42.09	43.68	36.01	20.44

Table 7: Macro-averaged accuracies for all words (%)

Table 6 indicates that the macro averaged accuracies of LDAWN (42.51%) outperformed those of SWSM2 (42.09%) when all the words were evaluated. However, the same table reveals that the reason is due to the results for the easy class words, i.e., the words that almost always had the same sense. In addition, Tables 5 and 6 indicate that SWSM clearly outperformed the other systems for words in the normal and hard classes.

9 Discussion

The examples“可能性 (possibility)” and “洗う (wash)” were cases where most senses were correctly predicted. “可能性 (possibility)” is a hard-class word and it appeared 18 times in the corpus. SWSM correctly predicted the senses of ∼70% of them. It had three senses as described in Section 1: (1) the ability to do something well, (2) its feasibility, and (3) the certainty of something happenings. First, SWSM could correctly predict the first sense. The words that surrounded them were, for instance, “両者 (both sides)” and “人間 (human)”, and “研究 (research)”, “コンビナート (industrial complex)”, and “今後 (hereafter)”. Second, SWSM could correctly predict almost none of the words that had the second sense. The words surrounding an example were “毎日 (every day)”, “違う (various)”, “直面する (to face)”, and “人々 (people)”, and SWSM predicted the sense as sense (1). We think that “人々 (people)” misled the answer. The words surrounding another example were “破る (break through)”, “音楽 (music)”, and “広げる (spread)”, and SWSM predict the sense as sense (1). We think that “広げる (spread)” could be a clue to predict the sense, but “音楽 (music)” misled the answer because it appeared many times in the corpus. Finally, SWSM could correctly predicted the last sense. The words surrounded them were, for instance, (1) “事態 (situation)”, “生ずる (arise)”, and “出る (appear)”, (2) “円高 (appreciation)”, “進む (escalate)”, and “出る (appear)”, and (3) “読む (read)” and“否定する (deny)”.

“洗う (wash)” is a normal-class word and it appeared five times in the corpus. SWSM correctly predicted the senses of ∼80%, viz., four of them. It has two senses in the corpus: (1) sanctify (someone’s heart) and (2) wash out a stain with water. The words surrounding the example that were incorrectly predicted were “今夜 (tonight)”, “体 (body)”, and “否 (not)”, and SWSM answered the sense as (1) even though it was (2). The words surrounding the examples that were correctly predicted were (1) “島民 (islander)”, “涙 (tear)”, and “石 (stone)”, (2) “見る (look at)” and “心 (heart)”, (3) “手足 (limb)”, “顔 (face)”, “私 (I)”, and “風呂 (bath)”, (4) “体 (body)”, “水 (water)”, and “抜く (drain)”.

These examples demonstrate that the surrounding words were good clues to disambiguate the word senses.

10 Conclusion

We proposed the surrounding word sense model (SWSM), which used the word sense distribution around ambiguous words, and performed unsupervised all-words word sense disambiguation in the Japanese language. The system incorporated the EDR concept dictionary and we performed experiments using the EDR Japanese corpus. We evaluated the performance of the model using difficulty classes based on the entropy of senses in the corpus: easy, normal, and hard. We performed experiments with SWSM in nine settings for the transition probability parameters. The experiments revealed that SWSM outperformed the random baseline and LDAWN, which is a system that uses the topic model. The SWSM model clearly outperformed the other systems for senses in the normal and hard classes. Some examples that correctly predicted senses indicated that the surrounding words were good clues to disambiguate word senses even if we used unsupervised WSD.

References

Timothy Baldwin, Su Nam Kim, Francis Bond, Sanae Fujita, David Martinez, and Takaaki Tanaka. 2008. Mrd-based word sense disambiguation: Further extending lesk. In *Proceedings of the 2008 International Joint Conference on Natural Language Processing*, pages 775–780.

David Blei, Andrew Ng, and Michael Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 1(3):993–1022.

Jordan Boyd-Graber, David M. Blei, and Xiaojin Zhu. 2007. A topic model for word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1024–1033.

W. Gale and G. Sampson. 1995. Good-turing smoothing without tears. *Journal of Quantitative Linguistics*, 2(3):217–237.

Weiwei Guo and Mona Diab. 2011. Semantic topic models: Combining word distributional statistics and dictionary definitions. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 552–561.

Hideki Hirakawa and Kazuhiro Kimura. 2003. Concept abstraction methods using concept classification and their evaluation on word sense disambiguation task. *IPSJ Journal*, 2(44):421–432, (In Japanese).

Jay J. Jiang and David W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of International Conference Research on Computational Linguistics*, pages 19–33.

Jun S Liu. 1994. The collapsed gibbs sampler in bayesian computations with applications to a gene regulation problem. *Journal of the American Statistical Association*, 427(40):958–966.

Diana McCarthy. 1997. Estimation of a probability distribution over a hierarchical classification. In *The Tenth White House Papers COGS - CSRP*, pages 1–9.

Rada Mihalcea. 2005. Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In *Proceedings of the 2005 Conference on Empirical Methods in Natural Language Processing*, pages 411–418.

Hideo Miyoshi, Kenji Sugiyama, Masahiro Kobayashi, and Takano Ogino. 1996. An overview of the edr electronic dictionary and the current status of its utilization. In *Proceedings of the COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*, pages 1090–1093.

Ted Pedersen, Satanjeev Banerjee, and Siddharth Patwardhan. 2005. Maximizing semantic relatedness to perform word sense disambiguation. In *Research Report UMSI*.

Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *International Joint Conferences on Artificial Intelligence*, pages 448–453.

Francesc Ribas. 1995. On learning more appropriate selectional restrictions. In *Proceedings of the Seventh Conference of the European Chapter of the Association for Computational Linguistics*, pages 112–118.

Yuto Sasaki, Kanako Komiya, and Yoshiyuki Kotani. 2014. Word sense disambiguation using topic model and thesaurus. In *Proceedings of the fifth corpus Japanese workshop*, pages 71–80 (In Japanese).

Kiyoaki Shirai. 2002. Construction of a word sense tagged corpus for senseval-2 japanese dictionary task. In *Proceedings of the third International Conference on Language Resources and Evaluation*, pages 605–608.